UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/708,397	11/08/2000	Roger Kenneth Abrams	RPS920000077US1	2446

45211 7590 12/02/2004

KELLY K. KORDZIK
WINSTEAD SECHREST & MINICK PC
PO BOX 50784
DALLAS, TX 75201

EXAMINER	
TRUONG, THANHNGA B	
ART UNIT	**PAPER NUMBER**
2135	

DATE MAILED: 12/02/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/708,397	ABRAMS, ROGER KENNETH	
	Examiner	Art Unit	
	Thanhnga Truong	2135	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *8/23/2004 (Appeal brief)*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-25* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-25* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *11 August 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. The Appeal Brief filed August 24, 2004 has been carefully considered by an Appeal Conference. The conferees agreed that Moran does teach the system using memory as show in Figure 1. However, Moran fails to teach the claimed memory being divided into pages, each page having an execution flag and a memory manager in the data processing system. Thus the finality of the office action mailed May 24, 2004 is now withdrawn. The office regrets any inconvenience due to the applicant.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-2, 9-12, 15-19, 24-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yates, Jr. et al (US 6,397,379).

a. *Referring to claim 1:*

i. Yates teaches:

(1) a bus system; a CPU connected to the bus system **[i.e., in general, in a forty-first aspect, the invention features a method and a microprocessor chip for performance of the method. As part of executing a stream of instructions, a series of memory loads is issued from a computer CPU to a bus, some directed to well-behaved memory and some directed to non-well-behaved devices in I/O space. (column 15, lines 5-10)]**;

(2) a RAM connected to the bus system, the RAM being divided into pages, each page having an execution flag; a memory manager configured to manage the pages of the RAM and permit CPU execution of data on pages according to the execution flag; a program stored within at least one page of the RAM; and a program stack stored within at least one page the RAM **[i.e., referring to Figure 1, in general, in a first aspect, the invention features a computer with an instruction**

processor designed to execute instructions of first and second instruction sets, a memory for storage of a program, a table of entries corresponding to the pages, a switch, a transition handler, and a history record. The memory is divided into pages for management by a virtual memory manager. The program is coded in instructions of the first and second instruction sets and uses first and second data storage conventions. The switch is responsive to a first flag value stored in each table entry, and controls the instruction processor to interpret instructions under, alternately, the first or second instruction set as directed by the first flag value of the table entry corresponding to an instruction's memory page. The transition handler is designed to recognize when program execution has transferred from a page of instructions using the first data storage convention to a page of instructions using the second data storage convention, as indicated by second flag values stored in table entries corresponding to the respective pages, and in response to the recognition, to adjust a data storage configuration of the computer from the first storage convention to the second data storage convention. The history record is designed to provide to the transition handler a record of a classification of a recently-executed instruction. (column 2, lines 28-51). In addition, particular embodiments of the invention may include one or more of the following features. The regions may be pages managed by a virtual memory manager. The indications may be stored in a virtual address translation entry, in a table whose entries are associated with corresponding virtual pages, in a table whose entries are associated with corresponding physical page frames, in entries of a translation look-aside buffer, or in lines of an instruction cache. The code at the first destination may receive floating-point arguments and return floating-point return values using a register-based calling convention, while the code at the second destination receives floating-point arguments using a memory-based stack calling convention, and returns floating-point values using a register indicated by a top-of-stack pointer.];

(3) wherein the memory manager is configured to determine whether the program is susceptible to buffer overflow attacks, and, if so, set

the execution flag for program stack pages of RAM to deny CPU execution of data on the program stack pages of RAM **[i.e., a limit detector is operatively interconnected with the register pointer to detect when a range of registers available for collecting profile information is exhausted, and a store unit is operatively interconnected with the limit detector of effect storing the profile information from the general registers to the main memory of the computer when exhaustion is detected. The profile circuitry comprises a plurality of storage registers arranged in a plurality of pipeline stages, information recorded in a given pipeline stage being subject to modification as a corresponding machine instruction progresses through the instruction pipeline. When an instruction fetch of an instruction causes a miss in a translation look aside buffer (TLB), the fetch of the instruction triggering a profileable event, the TLB miss is serviced, and the corrected state of the TLB is reflected in the profile information recorded for the profileable instruction. The profile control bits include a timer interval value specifying a frequency at which the profile circuitry is to monitor the instruction pipeline for profileable events. (column 10, lines 31-50). An alternative tuning method for TAXi_Control.Profile_Timer_Reload_Constant 494 considers buffer overruns. When the range of profile collection registers is full, the profile registers are spilled (536 and 548 of Figure 5a) to a ring buffer in memory. The hot spot detector 122 consumes the profile information from this ring buffer. If profiler 400 overruns hot spot detector 122 and the ring buffer overflows, then the value in TAXi_Control.Profile_Timer_Reload_Constant 494 is increased, to reduce the frequency at which profiling information is collected. Alternatively, on a buffer overrun, the frequency at which hot spot detector 122 runs can be increased (column 72, lines 9-19)].**

ii. Eventhough, Yates does not explicitly mention the use of excecution flag in the event of buffer overflow, Yates does disclose:

(1) In its most-common mode of operation, profiler 400 awaits a two-part trigger signal (516, 522 of Figure 5a) to start sampling events, and then records every profileable event 416 in a dense sequence, including every

profileable event that occurs, until it stops (for instance, on exhaustion of the buffer into which profile information is being collected), as opposed to a conventional profiler that records every n.sup.th event, or records a single event every n microseconds, whereby Figure 5a also shows the request flag and active flag are being used **(column 55, lines 5-12).**

iii. It would have been obvious to a person having ordinary skill in the art at the time the invention was made to:

(1) have applied these request and active flags to control buffer being overflow **(for more details see Figure 5 a state machine and column 65, line 63 through column 67, line 60).**

iv. The ordinary skilled person would have been motivated to:

(1) have applied these request and active flags to control buffer being overflow since the switch is responsive to a first flag value stored in each table entry, and controls the instruction processor to interpret instructions under, alternately, the first or second instruction set as directed by the first flag value of the table entry corresponding to an instruction's memory page **(column 36, lines 41).**

b. *Referring to claim 2:*

i. Yates further teaches:

(1) wherein the memory manager and the CPU are configured to deny CPU execution of data by triggering a hardware interrupt **[i.e., a fourth class includes asynchronous x86 transfers of control from hardware interrupts, page faults, breakpoints, single step, or any other x86 exception detected in converter 136 or emulator 316 that must be manifest to the x86 virtual machine (column 73, lines 20-24)].**

c. *Referring to claims 9, 11, 12, 15-18, 24-25:*

i. These claims have limitations that is similar to those of claim 1, thus it is rejected with the same rationale applied against claim 1 above.

d. *Referring to claim 10:*

i. This claim has limitations that is similar to those of claim 2, thus it is rejected with the same rationale applied against claim 2 above.

e. *Referring to claim 19:*

i. This claim has limitations that is similar to those of claims 1 and 2, thus it is rejected with the same rationale applied against claims 1 and 2 above.

4. Claims 3-8, 13-14, 20-23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yates, Jr. et al (US 6,397,379), and further in view of Wang (US 6,477,612 B1).

a. *Referring to claim 3:*

i. Yates teaches the claimed subject matter except for:

(1) an annotation API

ii. However, Wang teaches:

(1) APIs are provided in a memory management component of an operating system, to enable an application to request allocation of special amounts of virtual memory that are capable of being remapped to pages of physical memory, to request allocation of physical memory, and to control the mapping therebetween. The mapping may be to a fine granularity, as little as a single page in virtual memory mapped to a single page in physical memory. An API is also provided to free physical memory. The memory manager records the mappings in a table, which are then used in a processor's translation buffer to point a given virtual address to a mapped-to page of physical memory **(column 1, line 63 through column 2, line 8).** In addition, To request the allocation of a region of virtual memory 76, (e.g., the region 78 in Figure 3), as generally represented in Figures 3 and 4, a process 60.sup.1 makes an API function call to the memory manager 62. Via other APIs (described below) the process 60.sup.1 also requests the allocation of pages of physical memory 66, and then requests the mapping and remapping of addresses in virtual memory in one or more of the specially-allocated regions to the pages of the physical memory 66. Essentially, each mapping represents a pointer from a virtual address to a page in the physical memory 66. Note that although virtual memory 76 is not a tangible structure, it can be considered such from the viewpoint of the components (such as processes and the operating system) that access it, and is thus shown as a logically-organized structure in Figure 3 **(column 6, lines 8-22).**

iii. It would have been obvious to a person having ordinary skill in the art at the time the invention was made to:

(1) have applied an annotation API in Yates' computer system for enabling applications to access an increased amount of physical memory via an extension to virtual memory addressing. **(column 1, lines 61-63 of Wang).**

iv. The ordinary skilled person would have been motivated to:

(1) have applied an annotation API in Yates' computer system for managing computer system memory **(column 1, line 11 of Wang).**

b. *Referring to claims 4-5:*

i. These claims have limitations that is similar to those of claim 3, thus it is rejected with the same rationale applied against claim 3 above.

c. *Referring to claims 6-8:*

i. These claims have limitations that is similar to those of claims 1 and 3, thus it is rejected with the same rationale applied against claims1 and 3 above.

d. *Referring to claims 13 and 14:*

i. These claims have limitations that is similar to those of claim 3, thus it is rejected with the same rationale applied against claim 3 above.

e. *Referring to claims 20 and 21:*

i. This claim has limitations that is similar to those of claim 3, thus it is rejected with the same rationale applied against claim 3 above.

f. *Referring to claims 22 and 23:*

i. This claim has limitations that is similar to those of claims 3 and 9, thus it is rejected with the same rationale applied against claims 3 and 9 above.

Conclusion

5. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

a. Sober (US 6, 088,777) discloses a memory manager requests a large area of memory from an operating system, and from the viewpoint of the operating

system, that memory is fixed. That fixed memory area is then divided up into an integral number of classes, e.g. by the memory manager (see abstract).

b. Karkhanis et al (US 6,085,296) discloses a method of managing computer memory pages. The sharing of a program-accessible page between two processes is managed by a predefined mechanism of a memory manager. (see abstract).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Thanhnga (Tanya) Truong whose telephone number is 571-272-3858.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Vu can be reached at 571-272-3859. The fax and phone numbers for the organization where this application or proceeding is assigned is 703-872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 571-272-2100.

TBT
November 29, 2004

KIM VU
SUPERVISORY PATENT EX
TECHNOLOGY CENTER 2